

Intro to Next Generation Sequencing Workshop Botany 2010

Essential Linux

Compiled by Aaron Liston and Todd Mockler, Oregon State University

File and Directory Management

ls – List directory contents. **ll** – List directory contents and file details.

***** – Match any character in a directory or file name. For example, **ls *.fsa** lists all files that end with fsa.

mkdir – Make a directory.

cd – Change the current directory. Use **cd ..** to move up one directory.

cp – Copy files and directories. **mv** – Move (and rename) files and directories.

rm – Remove files and directories. By default, you must type **y** to approve each. You can turn this off with **rm -f**, but be very careful, as there is no recycle bin!

ln – Creates a link to a file. Used to access a single file from multiple locations.

Tab Key – Autocomplete names of files in a directory.

Up Arrow – Go back to previous commands.

Ctrl – u – Erase a command line.

dos2unix, mac2unix – Converts line return characters from Windows and Mac files to Linux.

chmod – By default, files can be read by all users, but only modified by their creator (you). To allow all other users to modify a file, type **chmod a+w filename (g+w** for only members of your group). To make a file executable (e.g. a shell script) type **chmod +x filename (g+x** for your group, and **a+x** for all users).

df -h – Reports how much disk space is available on your drive. **du -h** – Disk space available in a folder.

File Viewing

cat – Concatenate files and prints files to the screen (cat **-A** shows hidden characters, e.g. tabs).

less – The best way to view file contents (unless the file is very large).

head – Outputs the first part of files, by default the last ten lines (useful for large files).

tail – Outputs the last part of files, by default the last ten lines (use on output files to view progress).

wc – Prints the number of bytes, words, and lines in files.

q – In many programs, e.g. less, hitting 'q' will exit the program.

File Manipulation

& – Appending the & to the end of the command line will make the job run in the background.

> – Direct the output of a command to a file.

>> – Appends to an existing file instead of creating a new file.

| – Pipe the output of a command to another command.

grep – Searches a file (or standard input) for lines containing a match to a pattern. Examples:
grep ^ACGT filename > output will move all lines in “filename” that begin with ACGT to “output”.
grep -c ^ACGT will count the number of lines that begin with ACGT.

sed – Finds and replaces text in a file. Example:
sed s/A/T/g filename > output will replace all occurrences of A with T in “filename”.

sort - Sorts the lines of text files. A numeric sort for field (key) 3 is specified with **sort -n -k3**

uniq – Finds (and optionally counts) the unique lines in a sorted file. For example:
sort filename | uniq -c | sort -n -r > output
will sort and count the data in a file, then sort by reverse numeric order of the count.

cut – Copies one or more fields (by default, delimited by tabs) in a file.

File Compression and Compilation

gzip – To compress a single file. The compressed file(s) will be given the suffix **.gz**

gunzip – To extract a gzip compressed file.

tar – To combine and compress multiple files, or to separate and uncompress files. Used for installing programs such as Velvet: **tar -zxvf velvet-latest.tgz**

make – To compile a set of programs specified in a file named **makefile**.

Miscellaneous

history - Displays the command you have used in a session (very useful for documenting your work).

users – Displays other users on this computer.

top – Displays all jobs running on this computer.

ctrl+c – Terminate a program or job running in the foreground.

man – Displays a brief manual page for any command.
For example: **man cat** - shows the usage and options for the command ‘cat’:

info – Displays detailed manual pages for most commands.

There is an abundance of useful online information on Linux commands. If you are stuck, try Google!

Regular Expressions

Used to denote search terms that are not alphanumeric. Useful in commands such as **grep** and **sed**.
For example, in the **grep** command above, the **^** denotes “beginning of a line”. Likewise, **\$** denotes “end of a line”. A good introduction to regular expressions is <http://www.zytrax.com/tech/web/regex.htm>